

ISSN



9 772903 071009



Jurnal Teknik Informatika

Desember 2012

Vol. I #1

Rekursif

Created with




nitro PDF[®] professional
download the free trial online at nitropdf.com/professional

Rekursif

Jurnal
Teknik
Informatika

Volume 1 Nomor 1 Desember 2012


Penanggung Jawab
Ketua Program Studi
Teknik Informatika
Fakultas Teknik
UNIVERSITAS BENGKULU

Dewan Redaksi

Ketua :
Boko Susilo
Penyunting :
Elnawati
Arie Vitasari
Anggota :
Fanny Fanny Coasters
Rusdi Efendi

Reviewer

Ety Harnanthyah
Achaia Johar

Alamat Redaksi

Jurnal Teknik Informatika
Fakultas Teknik - Kampus Universitas Bengkulu
Jl. W.R. Supriatna Kuning
Limas Bengkulu 38271
Telp: 07365 34000, 31140 - 227
Email: wikin@unibengkulu.ac.id

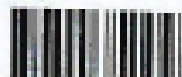
www.unibengkulu.ac.id

© Jurnal Teknik Informatika Rekursif

DAFTAR ISI

Pengantar Redaksi ii
Rancang Bangun Aplikasi Metode Tabu Search Pada Penyelesaian Assignment Problem 1-13
Boko Susilo, Septo Hastuti, Rusdi Efendi	
E-Learning Pengenalan Sibil (Sistem Isyarat Bahasa Indonesia) Untuk Anak Tunarungu Berbasis Multimedia 14-21
Boko S. Phony K.L, Dedi Andriyawan	
Aplikasi Sistem Pakar Untuk Mengidentifikasi Kerusakan Hardware Komputer Dengan Menggunakan Metode Forward Chaining Berbasis Web 22-34
Achaia Johar, Fanny F. Lodi, Cristiani	
Implementasi Algoritma CRC(Cyclic Redundancy Checks) untuk Deteksi Error dan Pengecekan Integritas File 35-45
Elnawati, Viki Ratihati, Dedi Andriyawan	
Perancangan Web Filtering Menggunakan Visual Basic 6.0 46-53
Anand Fekrisandira	

ISSN



9 772503 075009

Created with



nitroPDF[®]
Created with

professional
PDF[®]

download the free trial online at nitropdf.com/professional

download the free trial online at nitropdf.com/professional

IMPLEMENTASI ALGORITMA CRC (CYCLIC REDUDANCY CHECKS) UNTUK DETEKSI ERROR DAN PENGECEKAN INTEGRITAS FILE

¹Emawati, ²Vivik Pratiwi, ³Desi Andeswari,

Program Studi Teknik Informatika, Fakultas Teknik Universitas Bengkulu

ABSTRACT

Abstract – CRC (Cyclic Redundancy Checks) is detection technique which calculating bit-per-bit to bring out a value used to detect file error. CRC (Cyclic Redundancy Checks) has several variants based on polynomial number used in the process of computation which is CRC 16 and CRC 32. Moreover, CRC (Cyclic Redundancy Check) is also used as "finger print" used in identifying file integrity verification which is to prove that the file has not been modified or the file is still original. The objective of this research is to develop a software to detect error and file checking integrity in the process of data transfer by implementing Algorithm CRC. In this research, the system was developed using programming language Visual Basic 6.0 Enterprise Edition, Winetel development method, and Data Flow Diagram (DFD) system design. The result of this research has prove that Algorithm CRC (Cyclic Redundancy Checks) can be used to detect file error in the process of data transfer and CRC (Cyclic Redundancy Checks) Algorithm can also be use in the process of file integrity checking.

Key words: CRC (Cyclic Redundancy Checks), Error Detection, File Integrity

1. PENDAHULUAN

Proses transfer data terjadi antara 2 buah komputer yang saling terhubung yaitu komputer pengirim dan komputer penerima. Akan tetapi sering kali timbul pertanyaan bagaimana kondisi data tersebut setelah sampai pada komputer penerima. Data yang diterima oleh komputer penerima kadang kala berbeda dengan data yang dikirim oleh komputer pengirim. Hal ini bisa diakibatkan karena adanya gangguan-gangguan dalam proses transfer data, misalnya terdapat gangguan radiasi elektromagnetik, cakap silang (gangguan pada sinyal), petir atau

seperti adanya perubahan bit-bit data yang dikirimkan. Jika terjadi perubahan bit-bit data akan mengakibatkan data yang dikirimkan tersebut corrupt, sehingga data kadang tidak dapat di buka, karena data tersebut sudah mengalami kerusakan.

Proses pengiriman ataupun penyimpanan data seringkali mempunyai resiko terjadi perubahan yang tidak diinginkan terhadap data. Hal ini seringkali terjadi pada level fisik (media atau saluran yang digunakan), yang disebabkan karena gangguan (noise) pada proses penyimpanan ataupun pengiriman data itu sendiri. Untuk



mendeteksi kerusakan data ini, digunakan suatu cara untuk menghitung suatu nilai terhadap data yang diberikan, dan nilai tersebut dikirim bersama-sama data untuk dicak oleh penerima apakah data yang diterima sama dengan aslinya (tidak mengalami kerusakan selama perjalanan atau penyimpanan).

Dengan adanya gangguan pada proses transfer data, muncul suatu teknik untuk mendeteksi gangguan-gangguan tersebut. CRC (*Cyclic Redundancy Check*) merupakan suatu teknik deteksi yang dapat mengatasi kelemahan-kelemahan dari VRC dan LRC. CRC merupakan teknik deteksi yang melakukan perhitungan bit-per-bit untuk menghasilkan sebuah nilai, yang nantinya nilai tersebut dapat digunakan untuk deteksi error.

Algoritma CRC dapat diartikan sebagai suatu teknik perhitungan matematik dari suatu block data yang akan menghasilkan suatu nilai yang unik (checksum) yang berhubungan dengan data tersebut, sehingga nilai tersebut dapat digunakan untuk mendeteksi error. Untuk menjaga integritas data, maka kita menyimpan nilai CRC dari suatu data bersamaan dengan data tersebut (baik dalam proses transmisi maupun storage). Autentikasi dilakukan dengan melakukan pengecekan terhadap nilai CRC yang dihitung dari data yang diterima dengan nilai CRC yang disimpan untuk data tersebut. Jika nilai CRC-nya sama, maka

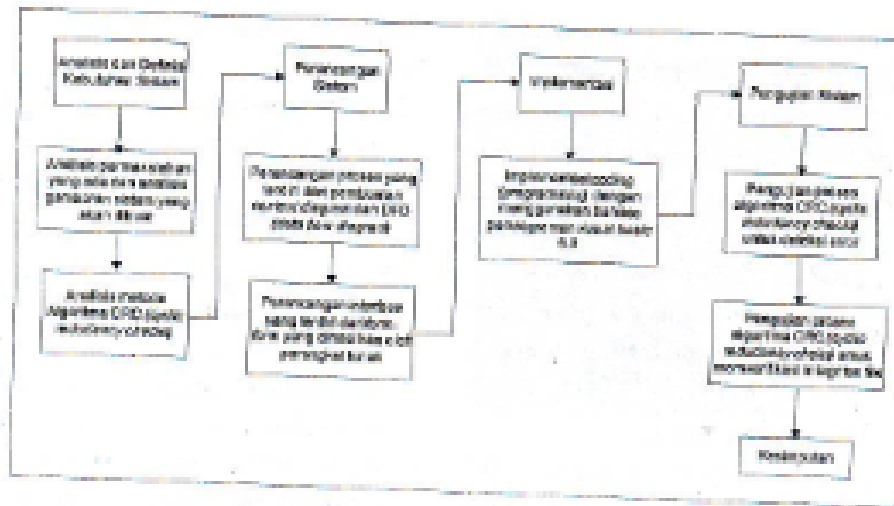
autentikasi berhasil (data tidak mengalami kerusakan selama proses transmisi ataupun storage).

Selain itu CRC juga dapat digunakan sebagai "fingerprint" yang dapat digunakan untuk melakukan identifikasi pembuktian integritas file, yaitu membuktikan bahwa file tersebut belum pernah dimodifikasi oleh siapapun atau file tersebut masih original. Pengecekan integritas file dilakukan dengan membangkitkan nilai dengan menggunakan algoritma CRC, kemudian nilai yang dihasilkan digunakan sebagai "finger print" atau suatu tanda yang digunakan untuk memverifikasi integritas file tersebut. Algoritma CRC cukup efektif digunakan karena mudah diimplementasikan dan cukup cepat dalam melakukan komputasinya. Oleh karena itu penulis membuat sebuah perangkat lunak dari implementasi algoritma CRC (*Cyclic Redundancy Check*) agar dapat memberikan manfaat kepada pengguna dalam mendeteksi error dan pengecekan integritas file dalam proses transfer data yang dilakukan.

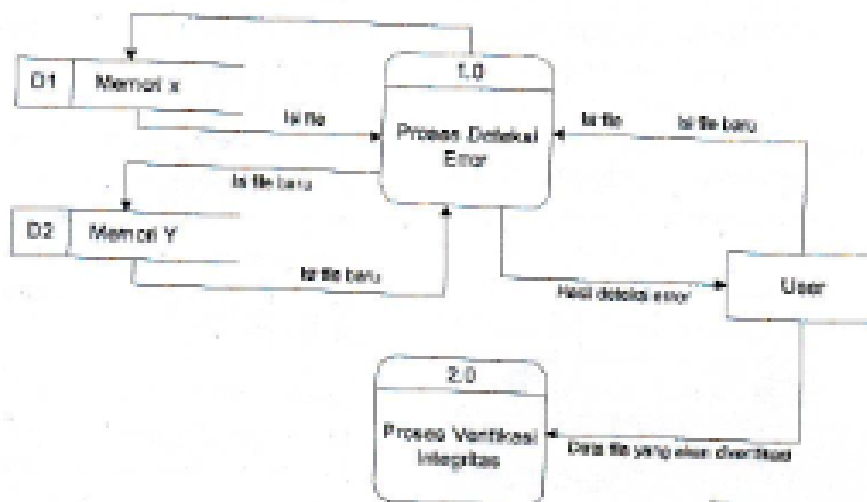
3. METODE PENELITIAN

Tahapan pekerjaan pada penelitian ini digambarkan dalam prosedur kerja pada Gambar 1, sedangkan untuk perancangan input, proses, dan output serta pengelolaan basis data digambarkan dengan DFD (*Data Flow Diagram*) pada Gambar 2.





Gambar 1. Prosedur Kerja Penelitian



Gambar 2. DFD Level-0

Tujuan utama dari transfer data adalah komputer penerima menerima suatu data yang sesuai dengan data yang dikirim oleh komputer pengirim. Akan tetapi sering kali terjadi perubahan bit-bit data yang

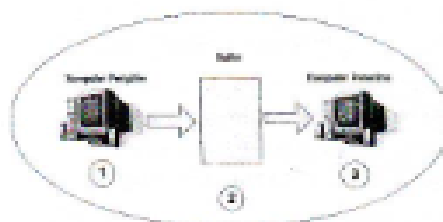
diterima tidak sama dengan bit-bit data yang dikirimkan. Hal ini dapat disebabkan adanya gangguan-gangguan pada saat pengiriman data ke komputer tujuan, sehingga dengan adanya permasalahan

tersebut adanya deteksi error memegang peranan yang sangat penting.

Untuk menyelesaikan permasalahan seperti yang telah dijabarkan, penulis akan membuat suatu program implementasi algoritma CRC untuk deteksi error dan pengecekan integritas file.

Program implementasi algoritma CRC untuk deteksi error dan pengecekan integritas file ini mampu menyelesaikan 3 permasalahan yaitu :

1. Membuktikan algoritma CRC mampu mendeteksi error terhadap suatu file yang ditransmisikan.
2. Mengukur unjuk kerja CRC-16 dan CRC-32 terhadap berbagai ukuran file yang sama.
3. Menerapkan algoritma CRC-32 untuk pengecekan integritas file.

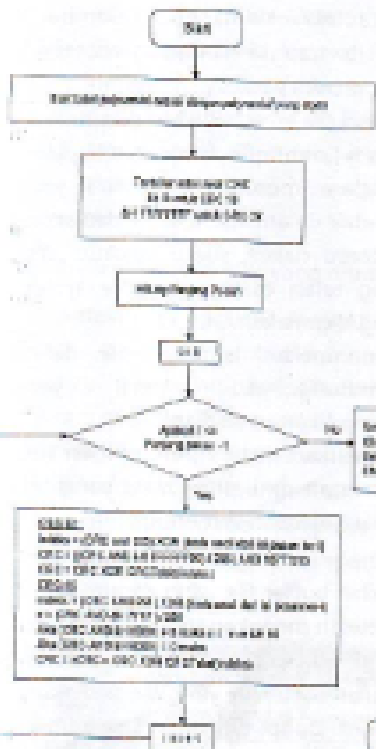


Gambar 3 Gambaran Umum Simulasi Program

Gambar 3 merupakan gambaran umum program simulasi untuk pemecahan masalah pertama. Adapun keterangan-keterangan dengan tanda nomor dalam

lingkaran dapat dipaparkan sebagai berikut :

1. Langkah pertama komputer pengirim mencari file yang akan dikirimkan ke komputer penerima kemudian dilakukan proses kirim.
2. Sebelum file yang dikirimkan sampai pada komputer penerima, file tersebut ditampilkan pada buffer dan isi dari file tersebut disimpan pada memori sementara misalkan X, dalam buffer terjadi proses perhitungan dengan menggunakan algoritma CRC untuk membangkitkan nilai checksum-32 dan checksum-16. Perhitungan masing-masing checksum dilakukan terhadap data yang ada pada memori sementara X tersebut. Lama proses perhitungan untuk masing-masing checksum diukur dengan menggunakan timer yang nantinya digunakan untuk menyelesaikan permasalahan yang kedua yaitu mengukur unjuk kerja CRC-16 dan CRC-32.
3. Setelah yakin untuk melanjutkan pengiriman file tersebut maka isi dari buffer akan kembali disimpan pada memori sementara misalkan Y, kemudian dapat melanjutkan proses pengiriman ke komputer penerima. Pada saat proses pengiriman terjadi perhitungan nilai checksum seperti langkah ke-2, akan tetapi perhitungan ini dilakukan untuk data yang baru, yaitu data yang terdapat pada memori



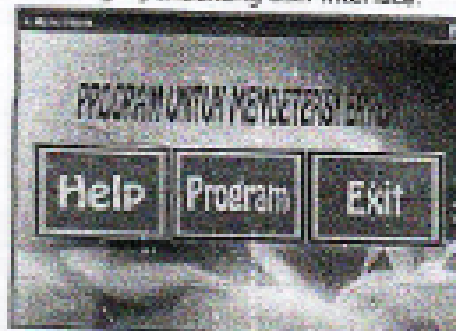
3. HASIL DAN PEMBAHASAN

3.1 Tahap Pengujian Program Deteksi Error

1. Masuk ke program untuk mendeteksi error

Pada saat pengguna masuk ke program untuk mendaftarkan email, pengguna pertama kali akan disajikan tampilan form utama. Form ini harus digunakan

sebagai pendukung user interface.

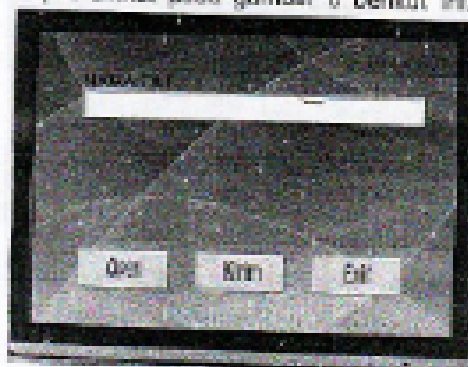


Gambar 5 Tampilan Form Utama Program Deteksi Error

Untuk menjalankan program deteksi error maka pengguna dapat mengklik tombol program. Setelah mengklik tombol program tersebut, akan tampil form tampilan komputer pengirim.

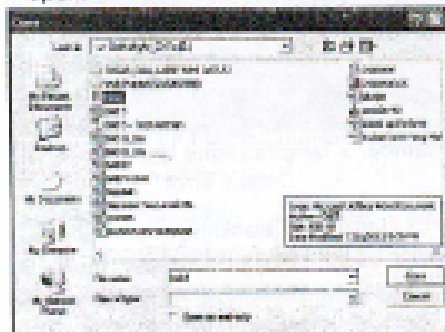
2. Tahap pemilihan file

Setelah pengguna mengklik tombol program pada tampilan form utama, maka pengguna akan dihubungkan ke tampilan komputer pengirim. Pada tampilan komputer pengirim ini terdapat 3 tombol yaitu tombol open, tombol kirim dan tombol exit. Tampilan komputer pengirim dapat dilihat pada gambar 6 berikut ini.



Gambar 6 Tampilan Komputer Pengirim

Gambar 6 menunjukkan bahwa komputer pengirim siap untuk memilih file yang akan ditransmisikan. Pertama-tama user pada komputer pengirim akan melakukan pemilihan file yang akan ditransmisikan dengan menekan tombol open.



Gambar 7 Hasil Tampilan Apabila Tombol Open Dipilih

Gambar 7 merupakan langkah untuk pemilihan file yang akan ditransmisikan. Dalam uji coba program, file yang akan ditransmisikan adalah "bab1.txt", pencarian file yang dipilih oleh user dilakukan oleh file system yang ada. Hasil tampilan setelah memilih file terdapat pada Gambar 8 berikut ini.



Gambar 8 Hasil Tampilan Sesudah Memilih File

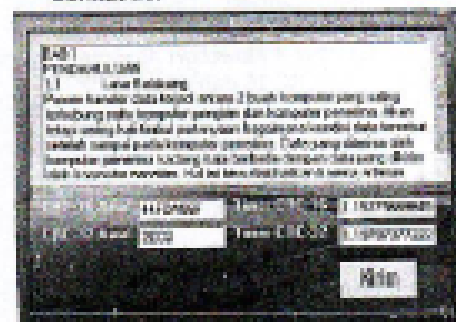
Setelah proses pemilihan file, proses

transmisi dapat dilanjutkan dengan menekan tombol kirim, seperti Gambar 8. Pada saat tombol kirim dijalankan terjadi beberapa proses yaitu :

1. Isi dari file tersebut akan ditampilkan dalam FormBuffer.htm dan dicopykan kedalam memori sementara yang terletak dalam suatu track dan sector tertentu dalam suatu struktur disk, yang telah diatur oleh file system yang digunakan.
2. Menampilkan isi dari file dalam FormBuffer.htm, seperti yang ditunjukkan pada Gambar 8
3. Melakukan proses perhitungan nilai checksum dari string data yang ada dalam memori sementara.

3. Tampilan buffer file yang dikirim

Setelah menekan tombol kirim pada form pengirim maka akan timbul hasil tampilan buffer file yang ditransmisikan tersebut. Dapat dilihat pada Gambar 9 berikut ini.



Gambar 9 Tampilan Buffer file yang dikirim

Pada Gambar 9 dapat dilihat bahwa buffer menampilkan :

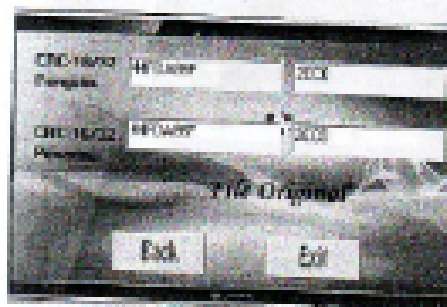
1. Isi file yang ditransmisikan yaitu "BAB

1 PENDAHULUAN*

2. Hasil CRC-16 dan CRC-32 sebelum file tersebut sampai di komputer pengirim yaitu CRC-16 Awal dan CRC-32 Akhir.
3. Waktu proses perhitungan CRC-16 dan CRC-32 yaitu timer CRC-16 dan timer CRC-32.

4. Hasil tampilan file yang ditransmisikan

Setelah pengguna mengklik tombol kirim pada form buffer, maka akan muncul hasil tampilan file tidak diedit ditransmisikan. Tampilan ini menerangkan bahwa file yang ditransmisikan tersebut tidak dilakukan pengeditan terhadap file, sehingga pada tampilan terdapat keterangan file original, seperti terlihat pada Gambar 10

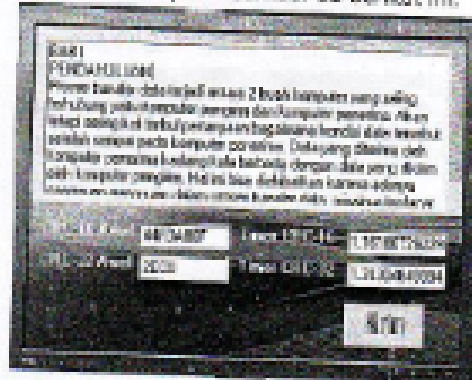


Gambar 10 File Tidak Diedit Ditransmisikan

Pada gambar 11 terlihat hasil perbandingan nilai CRC-16 dan CRC-32 pada komputer pengirim dan komputer penerima sama, maka dapat disimpulkan bahwa file yang diterima adalah "File Original"

5. Tampilan buffer file yang sudah diedit
6. Tampilan buffer yang sudah diedit

dapat dilihat pada Gambar 11 berikut ini.



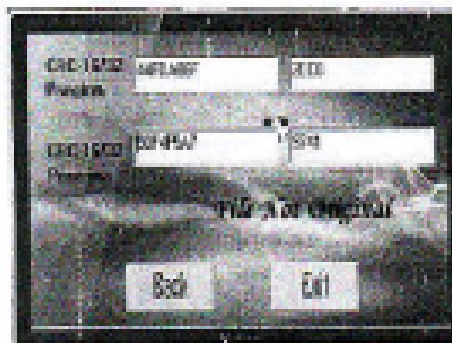
Gambar 11 Tampilan Buffer Yang Sudah Diedit

Untuk pembuktian deteksi error terhadap file yang ditransmisikan maka dilakukan proses edit terhadap isi file yang ditampilkan pada FormBuffer.frm, seperti yang ditunjukkan oleh Gambar 11 terdapat 2 kata yang dihapus pada isi file yaitu 1.1 latar belakang, sehingga pada saat tombol kirim pada buffer dilakukan, akan terjadi kembali 3 proses seperti proses sebelumnya yaitu :

1. Copy isi file yang ditampilkan oleh FormBuffer.frm ke dalam suatu memori sementara yang berbeda dengan memori_ sementara sebelumnya.
2. Proses perhitungan CRC dengan menggunakan isi file yang ada dalam memori sementara yang baru saja terbentuk.
3. Proses perbandingan CRC 32 file yang diterima dengan CRC-32 file yang dikirimkan.

6. Hasil tampilan file yang diedit ditransmisikan

Hasil tampilan file yang diedit menunjukkan bahwa file tersebut mengalami perubahan. Tampilan ini menerangkan bahwa file yang ditransmisikan tersebut telah dilakukan pengeditan terhadap file, sehingga pada tampilan terdapat keterangan *file not original*, seperti terlihat pada Gambar 12 berikut ini

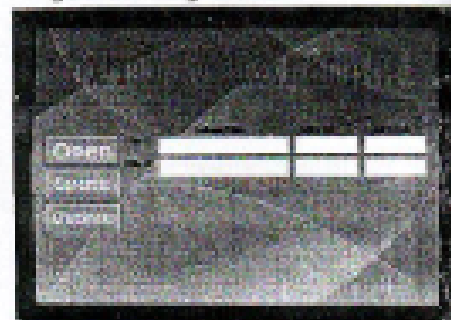


Gambar 12 File Yang Diedit Ditransmisikan

Perbandingan yang dilakukan digunakan untuk melakukan deteksi error terhadap file tersebut. Apabila hasil perbandingan tidak sama maka dapat disimpulkan bahwa file yang diterima adalah "File Not Original" (seperti pada gambar 12). Terlihat perbedaan antara nilai CRC-16 dan CRC-32 pada pengirim dan penerima berbeda atau file yang diterima tidak sama dengan file yang dikirimkan.

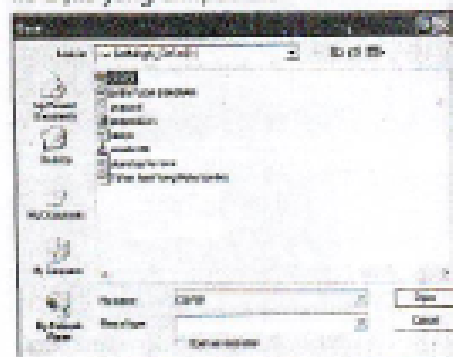
3.2 Hasil Pengujian Program

Pengecekan Integritas File



Gambar 13 Tampilan Layar Utama Pengecekan Integritas File

Gambar 13 merupakan tampilan layar utama untuk program pengecekan integritas file. Pada form ini terdapat 3 tombol yaitu tombol open untuk memilih file nama yang akan di generate atau dicek, tombol generate yang digunakan untuk menggenerate nilai checksum dari file yang diinputkan dan tombol checksum digunakan untuk mengecek integritas dari ke-2 file yang diinputkan.

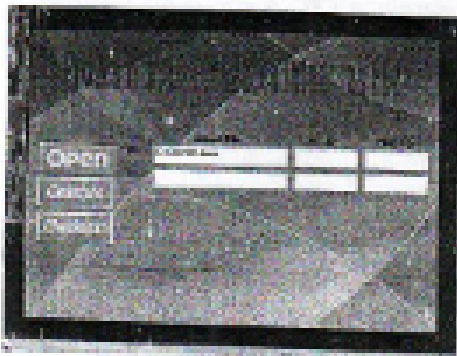


Gambar 14 Tampilan Open File

Untuk melakukan pembuktian terhadap program pengecekan integritas yang dilakukan pertama kali adalah

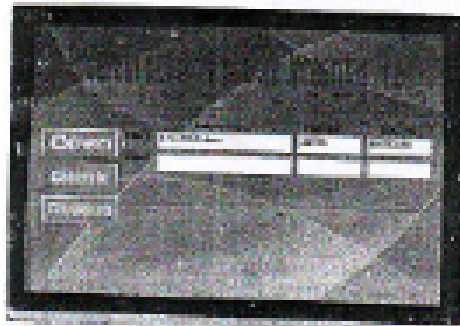


memilih file yang akan diverifikasi, dengan cara menekan tombol open, seperti pada gambar 14. (dalam percobaan ini menggunakan file cover.doc)



Gambar 15 Tampilan Setelah Memilih File

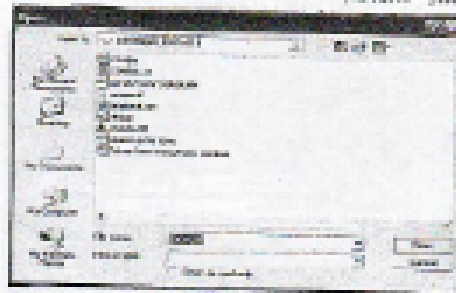
Gambar 15 merupakan hasil tampilan setelah memilih file yang akan di generate. File yang dipilih yaitu file cover.doc



Gambar 16 Tampilan Generate File

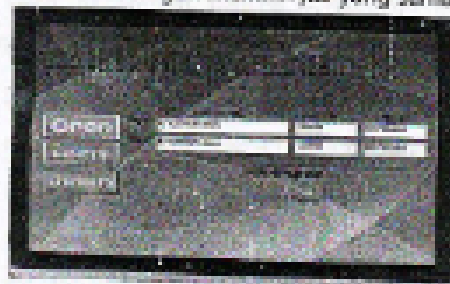
Setelah proses tersebut dilakukan maka proses selanjutnya adalah proses pada Gambar 16 yaitu meng-generate CRC-32, sebagai contoh file yang akan di-generate adalah cover.doc, pada saat tombol generate dipilih, maka terjadi perhitungan nilai checksum dari file cover.doc, pengaksesan dan pembacaan file cover.doc dilakukan

langsung menuju pada sector atau track dalam struktur disk dimana file tersebut disimpan dan diatur oleh sistem file.



Gambar 17 Tampilan Checksum

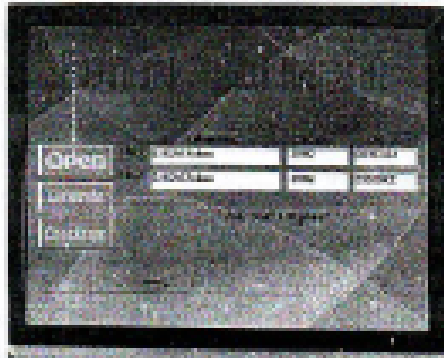
Setelah nilai checksum diperoleh maka nilai checksum tersebut disimpan dalam file terpisah yaitu cover.doc. File inilah yang berperan sebagai "finger print". Apabila tombol checksum dilakukan (Gambar 17) maka akan tampil kembali jendela open file, yaitu memilih file mana yang akan diperbandingkan dengan file yang di-generate sebelumnya, sehingga untuk memverifikasi integritas cover.doc, dapat dilakukan uji coba dengan cara melakukan checksum dengan memilih file yang sama



Gambar 18 Hasil Tampilan File Original

Pada Gambar 18 karena tidak melakukan perubahan sama sekali pada file tersebut maka hasil dari percobaan tersebut dapat dikatakan bahwa file tersebut masih

original dalam artian bahwa file tersebut, sama sekali tidak pernah diedit, ataupun dimodifikasi oleh siapapun.



Gambar 19 Hasil Tampilan File Not Original

Akan tetapi apabila terjadi perubahan pada file coverdoc, misalkan dengan cara melakukan save ulang terhadap file cover doc, maka hasil checksum yang dilakukan terhadap file tersebut adalah "file not original" seperti pada Gambar 19. hal ini terjadi karena adanya perubahan pada file coverdoc yaitu melakukan update terhadap header file tersebut, yang terjadi karena proses saving, sehingga dapat disimpulkan bahwa sekecil apapun perubahan yang terjadi terhadap suatu file dapat mempengaruhi integritas file tersebut, dan pengecekan terhadap file dibuktikan dengan menggunakan program pengecekan integritas file dengan menggunakan penerapan algoritma CRC.

4. KESIMPULAN DAN SARAN

Berdasarkan penyusunan naskah dan pembuatan program implementasi algoritma CRC (*cyclic redundancy check*) untuk deteksi error dan pengecekan integritas file, didapat beberapa kesimpulan yaitu :

1. Program implementasi yang dibuat ini membuktikan bahwa algoritma CRC (*cyclic redundancy check*) dapat digunakan untuk mendeteksi error file dalam proses transfer data yang dilakukan.
2. Algoritma CRC (*cyclic redundancy check*) juga dapat digunakan dalam proses pengecekan integritas file.

Berdasarkan hasil penelitian, pengujian serta pembahasan yang dibahas, maka penulis menyarankan untuk pengembangan penelitian dimasa yang akan datang sebagai berikut:

1. Setelah melakukan evaluasi terhadap aplikasi yang dibangun, penulis berharap agar aplikasi ini dapat dikembangkan lebih lanjut karena pada aplikasi yang dibuat, berupa simulasi program dalam 1 personal komputer. Untuk pengembangan aplikasi ini, tidak hanya berupa simulasi tetapi dapat diterapkan dalam kehidupan nyata.
2. Dapat dibuat program implementasi untuk mendeteksi virus dengan menggunakan algoritma CRC.



DAFTAR PUSTAKA

- [1] Anonim. 2007. *Cyclic Redundancy Check*. [Online]. Tersedia: http://en.wikipedia.org/wiki/Cyclic_redundancy_check.html. [4 November 2011].
- [2] Sedjati, Riza S. 2007. *Modul 4 Pendekatan Perancangan Teknikal dan Data Flow Diagram*. [Online]. Tersedia: http://www.unhas.ac.id/ikhsa/asnip/Aritektur_Komputer/sist_dan_analisis_sist/Microsoft_Word_-_Modul_4_APSI-Pendekatan_Perancangan_Tempdf [3 Maret 2012]
- [3] Saldi, Indra. 2009. *Penggunaan CRC32 dalam Integritas Data*. [Online]. Tersedia: <http://www.informatika.org/~rinaldi/Kriptografi/2006-2007/Makalah2/Makalah-054.pdf> [7 November 2011]
- [4] Zaini. 2009. *Pendeteksian dan Koreksi Error*. [Online]. Tersedia: http://file400142aini.NaskahPublikasi_Deteksi_error.pdf [30 Januari 2012]

